Perbandingan Efektivitas Algoritma Stemming dalam Dokumen Indonesia

Dyah Mustikasari[1] Ida Widaningrum[2,*] Rizal Arifin[3] Wahyu Henggal Eka Putri[4]

[1,2,3,4]*Jurusan Informatika, Universitas Muhammadiyah Ponorogo, 63471, Indonesia*
[*]*Penulis yang sesuai. Surel: iwidaningrum@umpo.ac.id*

ABSTRAK

Stemming merupakan suatu proses penentuan kata dasar dengan beberapa aturan. Dalam Bahasa Indonesia caranya adalah dengan menghilangkan prefiks, sisipan, sufiks, atau gabungan prefiks dan sufiks pada kata turunan. Beberapa algoritma stemming untuk Bahasa Indonesia telah dikembangkan. Namun efektivitasnya belum diteliti. Pada penelitian ini ketiga algoritma stemming tersebut akan dibandingkan. Kami menggunakan 900 imbuhan untuk melakukan perbandingan. Setiap kata dicari kata dasarnya menggunakan ketiga algoritma tersebut. Kata dasar yang dihasilkan kemudian dirujuk ke KBBI atau kamus bahasa Indonesia untuk mengetahui kebenarannya. Proses perbandingan stemming menunjukkan bahwa Sastrawi mampu melakukan stemming terbaik dengan 95,2% kata imbuhan yang diuji dapat menjadi kata dasar. Algoritma Nazief & Adriani menghasilkan 92,4%, sedangkan Arifin Setiono mencapai 89%. Bisa dikatakan, tulisan Arifin Setiono perlu banyak perbaikan karena banyak kata imbuhan yang tidak bisa kembali ke kata dasar.

Kata kunci:*Efektivitas, Stemming, Bahasa Indonesia, Dokumen.*

1. PERKENALAN

Stemming merupakan suatu proses mengembalikan kata imbuhan ke kata dasar atau akar kata dengan menggunakan aturan yang telah ditentukan. Caranya adalah dengan menghilangkan prefiks, infiks (penyisipan), sufiks, dan konfiks (gabungan awalan dan akhiran) dari kata imbuhan. Stemming merupakan bagian penting dalam Information Retrieval untuk pencarian web, pengelompokan dokumen dalam hal mengurangi jumlah indeks berbeda dari suatu dokumen, dan penerjemahan [1]. Stemmer atau algoritma yang baik akan mengembalikan kata imbuhan ke kata dasar dengan benar. Dalam bahasa Indonesia, awalannya adalah*pe-, saya-, ber-, di-, ke-, ter-,*dan konfigurasinya adalah *ke-an, ber-an, pe-an, se–nya*. Misalnya, kata*Baca* artinya 'membaca', bisa diberi awalan*Saya-* perubahan itu menjadi*membaca,*bukan*mebaca*. Hal ini menyebabkan adanya aturan perubahan fonem dalam pembentukan kata imbuhan dalam Bahasa Indonesia [2]. Contoh lainnya, kata*pukul* akan*memukul*jika itu menambahkan awalan*Saya-*, bukan*mempukul*. Fonem /p/ pada kata tersebut*pukul*menghilang. Namun, ketika kata tersebut*proses*ditambahkan awalan*Saya-*, itu menjadi*memproses*, bukan*memrose*. Dalam hal ini fonem /p/ pada kata tersebut*proses* tetap dipertahankan dan tidak hilang.

Beberapa aturan stemming telah dikembangkan untuk Bahasa Indonesia, seperti Algoritma Nazief & Adriani (1996) yang kemudian diselesaikan oleh Jelita Asian (2005) [1] , Vega Bressan [1], Arifin Setiono Algoritma (2002) [3], dan yang terbaru adalah Stemmer karya Sastrawi [4].

Algoritma Nazief & Adriani diusulkan oleh Bobby Nazief dan Mirna Adriani [5]. Algoritma ini menggunakan aturan morfologi dalam Bahasa Indonesia. Mereka dikumpulkan menjadi satu dan dikemas dalam afiks yang diperbolehkan dan afiks yang tidak diperbolehkan [1]. Algoritma Arifin Setiono memiliki proses yang mirip dengan algoritma Nazief & Adriani namun diasumsikan bahwa sebuah kata mempunyai dua awalan dan tiga akhiran [3]. Sedangkan Algoritma Sastrawi menerapkan algoritma berbasis Nazief-Adriani, kemudian disempurnakan dengan Algoritma CS (Confix Stripping), dan disempurnakan dengan algoritma ECS (Enhanced Confix Stripping), dan ditingkatkan lagi dengan Modified ECS.

Dengan berbagai metode stemming tersebut, belum diketahui algoritma mana yang terbaik untuk mengembalikan kata imbuhan ke kata dasar dalam Bahasa Indonesia. Beberapa studi banding telah dilakukan sebelumnya. Salah satu perbandingan yang pernah dilakukan adalah [6] yang membandingkan Algoritma stemming antara Nazief-Adriani, Arifin-Setiono, Tala, dan Vega. Perbandingan algoritma stemming lainnya dilakukan oleh [3] yang membandingkan antara Porter dan Arifin-Setiono [7]. Perbandingan Nazief-Adriani dan Idris juga dilakukan oleh [8]. Studi perbandingan yang dilakukan oleh [5] menguji antara beberapa algoritma yaitu porter confix striping, Nazief, Arifin,

Fadillah, Asian, Enhanced confix stripping, dan Arifiyanti [9]. Algoritma Sastrawi belum pernah dibandingkan pada penelitian-penelitian sebelumnya. Oleh karena itu pada penelitian ini akan dilakukan perbandingan antara Algoritma Nazief-Adriani, Algoritma Arifin-Setiono dan Sastrawi.

2. METODE

2.1.Nazief-Adriani

Algoritma Nazief & Adriani menggunakan kamus akar kata yang dibuat oleh Bobby Nazief dan Mirna Adriani. Algoritme memiliki langkah-langkah berikut:

1. Memeriksa apakah kata tersebut ada dalam kamus akar kata. Jika ditemukan maka dianggap sebagai kata dasar, kemudian proses dihentikan.
2. Menghilangkan akhiran infleksi (“-lah”, “-kah”, “-nya”, “-mu”, atau “-ku”). Jika dilakukan, maka akhiran tersebut merupakan partikel “-kah” dan “-lah”, langkah ini diulangi untuk menghilangkan akhiran kata ganti posesif (“-ku”, “-mu”, “-nya”)
3. Menghapus akhiran (“-i” atau “-an”). Jika akar kata ditemukan maka proses dilanjutkan ke 4. Jika tidak ditemukan maka langkah dilanjutkan ke 3a.
   A. Jika huruf terakhir adalah “-k”, maka huruf “-k” dihilangkan dan lakukan langkah 4. Namun jika akar kata masih belum ditemukan, maka lanjutkan ke langkah 3b.
   B. Akhiran yang dihapus (“-i”, “-an” dan juga “-kan”) dikembalikan, kemudian dilanjutkan ke langkah 4
4. Menghapus awalan ("be -", "di-", "me -", "ke -", "pe -", "te-" dan "se-"). Jika kata tersebut cocok dengan kamus akar kata, proses dapat dihentikan. Namun jika kata dasar belum ditemukan, maka kata tersebut dikodekan ulang. Proses ini dapat dihentikan jika:
   A. Kombinasi awalan dan akhiran yang salah.
   B. Awalan yang terdeteksi dengan awalan yang dihilangkan sebelumnya adalah sama.
   C. Menghapus tiga awalan.
5. Jika semua langkah sudah dilakukan namun kata dasar tidak dihasilkan, maka algoritma akan mengembalikan kata seperti sebelum stemming.

Nazief & Adriani kemudian diselesaikan oleh Asia [1]. Keunggulan Jelita Asian adalah:

1. melengkapi kamus,
2. menambahkan aturan untuk bentuk jamak (seperti *buku-buku*, *berbalasan-balasan*, dan seterusnya),
3. menambahkan awalan dan akhiran seperti *-permainan kata-kata*, mengubah kondisi untuk awalan *ter-, pe-, mem-, meng-*
4. mengubah urutan stemming kata imbuhan dengan awalan *ber-*dan akhiran *-lah*, awalan *ber-*dan akhiran *-sebuah*, awalan *Saya-*dan akhiran *-Saya,* awalan *di-*dan akhiran *-Saya*, awalan *pe-*dan akhiran *-Saya,* dan awalan *ter-*dan akhiran *-Saya,*untuk menghapus awalan terlebih dahulu dan kemudian akhiran.

2.2.Algoritma Arifin-Setiono

Pada tahun 2002, Agus Zainal Arifin dan Ari Novan Setiono mengajukan beberapa aturan untuk mengembalikan kata imbuhan ke kata dasar, yang kemudian dikenal dengan Algoritma Arifin-Setiono [3]. Algoritma ini mengasumsikan bahwa setiap kata mempunyai dua prefiks dan tiga sufiks, kemudian mengikuti pola berikut:

AW 1 + AW 2 + KD + AK 3 + AK 2 + AK 1

yang mana:

SEBUAH 1	= Awalan 1
SEBUAH 2	= Awalan 2
KD	= akar kata
AK 1	= Akhiran 3
AK 2	= Akhiran 2
AK 3	= Akhiran 1

Jika suatu kata memiliki awalan atau akhiran kurang dari itu, maka untuk awalan kosong diberi tanda x dan xx untuk akhiran kosong. Stemming dilakukan dengan urutan sebagai berikut:

A. pertama dikeluarkan AW 1, kemudian hasilnya disimpan di p1

B. lalu keluarkan AW 2, sehingga hasilnya tersimpan di p2

C. lalu keluarkan AK 3 agar hasilnya tersimpan di s1

D. kemudian melepas AK 2, dan hasilnya disimpan di s2

e. terakhir keluarkan AK 1, lalu hasilnya disimpan di s3

Hasil pemotongan tiap urutan dicocokkan dengan kamus apakah sudah kembali ke akar kata. Jika sudah kembali ke akar kata, maka stemming dihentikan, jika tidak maka proses dilanjutkan. Jika akar kata belum ditemukan dalam kamus hingga akhir rangkaian stemming, maka hasilnya digabungkan dengan imbuhan dengan menggunakan 12 konfigurasi berikut:

A. KD

B. KD + AK 3

C. KD + AK 3 + AK 2

D. KD + AK 3 + AK 2 + AK 1

e. AW 1 + AW 2 + KD

F. AW 1 + AW 2 + KD + AK 3

G. AW 1+ AW 2 + KD + AK 3 + AK 2

H. AW 1+ AW 2 + KD + AK 3 + AK 2 +AK1

Saya. AW 2 + KD

J. AW 2 + KD + AK 3

k. AW 2 + KD + AK 3 + AK 2

aku. AW 2 + KD + AK 3 + AK 2 + AK 1

Aturan-aturan ini ditulis dengan python yang dapat ditemukan di dalam itu mengikuti tautan: http://tiny.cc/stemmingarifinsetiono

2.3.Sastrawi

Sastrawi sebenarnya adalah perpustakaan stemmer. Library ini tersedia pada situs penyedia source code dan dapat diakses pada link https://github.com/sastrawi/sastrawi. [4] mengulas bahwa perpustakaan ini berdasarkan penelitian dari [1] [5] [10]. Ditulis di situsnya bahwa proses stemming menggunakan stemmer ini sangat bergantung pada kamus akar kata. Ini menggunakan kamus kata dasar dari kateglo.com dengan sedikit perubahan. Aturan stemmer Sastrawi adalah sebagai berikut :

A. Pertama adalah memeriksa apakah kata yang akan di-stemmed ada pada kamus kata dasar atau tidak. Jika ada, maka proses akan berhenti pada langkah ini.

B. Apabila kata tersebut tidak ada dalam kamus, berarti merupakan kata imbuhan, maka akhiran tersebut dihilangkan- *lah, -kah, -ku, -mu, -nya*,*-lah, -kah, -tah*atau-*permainan kata-kata*.

C. Menghapus imbuhan turunan-*aku, -kan, -an,*lalu menghapus*menjadi-, di-, ke-, aku, pe-, se-*Dan*te-*.

D. Apabila akar kata hasil langkah sebelumnya tidak ditemukan dalam kamus, maka dilakukan pengecekan apakah kata tersebut termasuk dalam tabel ambigu pada kolom terakhir atau tidak.

e. Akhirnya, ketika semua langkah di atas gagal, algoritme mengembalikan kata tersebut ke kata aslinya.

Semua algoritma di atas menggunakan kamus akar kata yang dapat diakses di http://tiny.cc/rootwords.

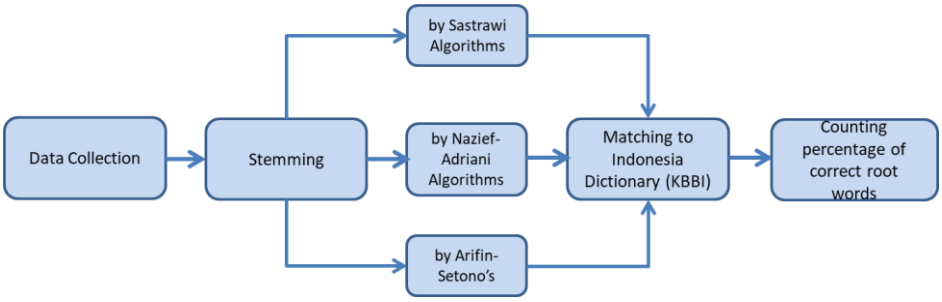
2.4.Perbandingan

Kami mulai dengan mengumpulkan data yang diuji. Terdapat 900 afiks dalam Bahasa Indonesia yang akan diuji yang dapat diakses pada link http://tiny.cc/dataset_stemming. Tabel 1 menggambarkan 50 afiks yang digunakan untuk stemming dengan algoritma Sastrawi, Nazief-Adriani, dan algoritma Arifin Setiono.

Tabel 1. Contoh Kata yang Dibubuhkan

imbuhan				
berantai	memakai	ajaran	kebijakan	perkenalan
bajakan	mengunjungi	dinyatakan	siapkah	perampasan
perkenalan	pegangan	dirampas	terasing	perenang
berurutan	mengenakan	ditunjukkan	teratur	terimalah
lebih baik	serapan	pemancaran	keberagaman	ukuran
mungkin	terimalah	pembentukan	meskipun begitu	ulangan
pemangkasan	teringat	pemberian	pernikahan	pembangkit
pemadam	terinjak	penggelapan	diperhatikan	kecepatam
mengepalai	teriris	pengikut	perencana	diselami
menikah	terbitan	penjagaan	mengarang	berakhirnya

Kata-kata tersebut dikumpulkan secara acak dari Kamus Bahasa Indonesia. Setiap kata dicari kata dasarnya menggunakan ketiga algoritma tersebut. Hasil dari proses stemming tersebut dirujuk ke Kamus Besar Bahasa Indonesia atau Kamus Besar Bahasa Indonesia untuk mengecek benar atau tidaknya. Mereka juga diperiksa secara manual untuk memastikan kembali ke akar kata sesuai konteks kata. Metodenya digambarkan secara singkat pada Gambar 1.



Gambar 1 Urutan penelitian.

3. HASIL DAN PEMBAHASAN

Kami menggunakan 900 afiks untuk melanjutkan stemming. Setiap algoritma telah mengembalikan kata dasar dengan beberapa kesalahan. Sastrawi melakukan 43 kesalahan dalam proses stemming atau mampu mengembalikan 857 akar kata yang benar. Tabel 2 menunjukkan sepuluh kesalahan.

Meja 2. Contoh Akar Kata yang Salah Diperoleh Sastrawi

TIDAK.	Kata-kata yang ditempelkan	KBBI	Sastrawi
1	berantai	rantai	beranta
2	bajakan	bajakan	baja
3	perkenalan	kenal	akhir
4	berurutan	urut	rurut
5	lebih baik	terbang	bangan
6	mungkin	bukan	mungkin
7	pemangkasan	pangkas	mangga
8	pemadam	padam	nyonya
9	mengepalai	kepala	palai
10	menikah	nikah	meni

Algoritma Nazief-Adriani mampu mengembalikan 832 imbuhan yang benar sesuai KBBI. Artinya ada 68 kesalahan yang dilakukan Nazief-Adriani dalam proses stemming. Tabel 3 mengilustrasikan beberapa kesalahan.

Tabel 3. Contoh Akar Kata yang Salah Diperoleh oleh Nazief-Adriani

TIDAK.	Kata-kata yang ditempelkan	KBBI	Nazief-Adriani
1	berantai	rantai	anta
2	bajakan	bajakan	baja
3	kebijakan	bijak	bija
4	memakai	pakai	maka
5	mengunjungi	ayolah	mengunjungi
6	mungkin	bukan	bu
7	pemangkasan	pangkas	mangga
8	pemadam	padam	nyonya
9	pegangan	pegang	gang
10	menikah	nikah	meni

Ada beberapa kata yang mempunyai imbuhan sama tetapi berasal dari akar kata yang berbeda, misalnya kata “mengepak” kata ini dapat berasal dari akar kata “epak” (mengambil hak atas sesuatu yang menghasilkan hasil dengan membayar sewa atau pajak) dan “ kepak” (mengepak). Menurut KBBI, jika kedua kata tersebut ditambah awalan “me-”, maka akan berubah menjadi kata “mengepak”. Contoh lain, kata “acau” (berbicara saat tidur) dan “kacau” (berantakan), jika diberi awalan “me-”, menjadi kata “mengacau”. Hal ini menunjukkan bahwa proses stemming berdasarkan pengecekan kamus akar, mengembalikan imbuhan pada akar kata pada urutan pertama. Jadi untuk imbuhan “mengacau” akan selalu kembali ke akar kata “acau”. Hasil keseluruhan dari proses stemming ketiga algoritma tersebut dapat dilihat pada link http://tiny.cc/result_stemming. Dari hasilnya, kami menghitung persentase akar kata yang benar dengan:

$$\frac{\text{Kata dasar yang benar}}{\text{Himpunan data}} \times 100\%$$

Kata dasar yang benar merupakan hasil stemming yang dicocokkan dengan kata dalam Kamus Besar Bahasa Indonesia atau KBBI. Tabel 5 menunjukkan persentase ketiga algoritma.

Sedangkan imbuhan yang dapat dikembalikan ke akar kata yang benar dengan algoritma Arifin-Setiono sebanyak 801. Terdapat 99 akar kata yang tidak sesuai dengan KBBI. Sepuluh kesalahan Algoritma Arifin-Setiono ditunjukkan pada Tabel 4.

Tabel 4. Contoh Akar Kata yang Salah Diperoleh Arifin-Setiono

TIDAK.	Kata-kata yang ditempelkan	KBBI	Arifin-Setiono
1	berantai	rantai	berantai
2	bajakan	bajakan	baja
3	kebijakan	bijak	bija
4	memakai	pakai	maka
5	berurutan	urut	berurutan
6	mungkin	bukan	mungkin
7	mengenakan	kena	enak
8	pemadam	padam	nyonya
9	pegangan	pegang	gang
10	serapan	serap	rap

Tabel 5. Persentase akar kata yang benar dari proses stemming

Algoritma	Benar Akar Kata	Salah Akar Kata	Persentase benar akar kata
Sastrawi	857	43	95,2%
Nazief-Adriani	832	68	92,4%
Arifin-Setiono	801	99	89%

4. KESIMPULAN

Hasil tersebut menunjukkan bahwa hasil tertinggi dari ketiga algoritma stemming adalah stemmer Sastrawi, disusul Nazief-Adriani. Dapat juga dikatakan bahwa Algoritma Arifin Setiono perlu banyak perbaikan karena banyak kata yang dibubuhi tidak dapat kembali ke kata dasar.

PENGAKUAN

Kami mengucapkan terima kasih kepada Direktorat Riset dan Pengabdian Masyarakat, Direktorat Jenderal Penguatan Riset dan Pengembangan Kementerian Riset, Teknologi, dan Pendidikan Tinggi Republik Indonesia, Kementerian Riset dan Teknologi.

REFERENSI

[1] J. Asian, HE Williams, dan SMM Tahaghoghi, "Stemming Indonesian," di *Konferensi Ilmu Komputer Australasia ke-28 (ACSC2005)*, 2005, jilid. 38.

[2] R. Setiawan, A. Kurniawan, W. Budiharto, dan A. Awalan, "Klasifikasi Afiks Fleksibel untuk Stemming Bahasa Indonesia," dalam *Konferensi Internasional ke-13 Teknik Elektro/ Elektronik, Komputer, Telekomunikasi dan Teknologi Informasi (ECTI-CON)*, 2016.

[3] A. Zainal dan A. Novan, "Klasifikasi Dokumen Berita Kejadian Berbahasa Indonesia dengan Algoritma Single Pass Clustering," di *Prosiding Seminar Teknologi Cerdas dan Penerapannya (SITIA)*, 2002.

[4] U. Hasanah, T. Astuti, R. Wahyudi, Z. Rifai, dan R. A. Pambudi, "Studi Eksperimental Teknik Preprocessing Teks untuk Penilaian Jawaban Singkat Otomatis dalam Bahasa Indonesia," dalam *Konferensi Internasional ke-3 tentang Teknologi Informasi, Sistem Informasi dan Teknik Elektro (ICITISEE)*, 2018, hlm.230–234.

[5] B. Nazief dan M. Adriani, "Confix stripping: Pendekatan Algoritma Stemming untuk Bahasa Indonesia," Jakarta, 1996.

[6] MSH Simarangkir, "Studi Perbandingan Algoritma-Algoritma Stemming untuk Dokumen Teks Bahasa Indonesia," *J.Infokar*, jilid. 1, tidak. 1, hal. 40–46.

[7] D. Novitasari, "Perbandingan Algoritma Stemming Porter dengan Arifin Setiono untuk Menentukan Tingkat Ketepatan kata Dasar," vol. 1, tidak. 2, hal.120–129, 2016.

[8] A. Prasdhata and KM Suryaningrum, "Perbandingan Algoritma Nazief & Adriani Dengan Algoritma Idris Untuk Pencarian Kata Dasar," *J.Teknol. dan Manaj. Memberitahukan.*, jilid. 4, tidak. 1, hal. 1–4, 2018.

[9] AS Rizki, "Perbandingan Stemmer Bahasa Indonesia dan Dampaknya pada Penggalian Teks Bahasa Indonesia, Studi Kasus Pengelompokan Keluhan Pelanggan PLN," Institut Teknologi Sepuluh Nopember, 2017.

[10] AZ Arifin dan HT Mahendra, I Putu Adhi Kerta Ciptaningtyas, "Enhanced Confix Stripping Stemmer dan Algoritma Semut untuk Klasifikasi Dokumen Berita Berbahasa Indonesia," di *Konferensi Internasional ke-5 tentang Teknologi dan Sistem Informasi & Komunikasi*, 2007, hlm.149–158.